

¿Por qué no se lanzó la excepción?

José está viendo el entrenamiento de **Java parte 4: Entendiendo excepciones** y, siguiendo al instructor, decidió crear la clase Flujo, para ver cómo funciona el lanzamiento de excepciones.

La clase tiene el método *main* y otros dos métodos, siendo en el segundo, José instancia una `ArithmeticException`:

```
public class Flujo {  
  
    public static void main(String[] args) { System.out.  
        System.out.println("Fin del main");  
    }  
  
    private static void metodo1() {  
        System.out.println("Inicio del metodo1");  
        metodo2();  
        System.out.println("Fin del metodo1");  
    }  
  
    private static void metodo2() {  
        System.out.println("Inicio del metodo2");  
        ArithmeticException exception = new ArithmeticEx  
    }  
  
}
```

[COPIA EL CÓDIGO](#)

Sin embargo, cuando José ejecutó la clase, se sorprendió, ya que no se lanzó la excepción. Ayuda a José seleccionando la alternativa que resuelve el problema.

A

Al crear una instancia de *ArithmeticException*, José se olvidó de pasar *metodo2* por parámetro:



```
private static void metodo2() {  
    System.out.println("Inicio del metodo2");  
    ArithmeticException exception = new ArithmeticException();  
}
```



José instancia la excepción, pero no lo lanzo a través del *throw*:



```
private static void metodo2() {  
    System.out.println("Inicio del metodo2");  
    ArithmeticException exception = new ArithmeticException();  
    throw exception;  
}
```

Correcto. No es suficiente crear una instancia de la excepción, es necesario lanzarlo a través del *throw*.

C

José instancia la excepción, pero no lo lanzo a través del *raise*:



```
private static void metodo2() {  
    System.out.println("Inicio del metodo2");  
    ArithmeticException exception = new ArithmeticException();  
    raise exception;  
}
```

PRÓXIMA ACTIVIDAD

